



第1章 分散並列型 動的解析システムについて

1.1 はじめに

システムSPACE (Space frame Analysis package for Civil Engineers, researchers and students) Ver.3.00は、幾何学的非線形を考慮した弾塑性解析システムであり、スペースフレームの静的・動的解析を効率よく実行し、また、その結果を処理し、グラフ、表、アニメーション等を用いて効果的に各種の情報を提供する。さらに、複雑で時間のかかる処理を実行するために、分散並列型の動的解析システムを新たに開発し、プロトタイプとして、Ver.1.10をシステムSPACEに組み込んだ。

このマニュアル：分散並列型動的解析編は、PC クラスターを利用した分散並列型動的解析システムについての解説書である。この分散並列型動的解析システムは、SPACE システムに組み込まれ、SPACE から直接利用可能となっている。ここでは、このシステムの利用方法、動的解析手法、並びに、各 PC でデータを共有するための通信手法について解説する。特に、独自に開発した MPI 準拠の並列用通信ライブラリーMPI_c (message-passing-interface_compact)について、全てのコードを記載し、解説する。

分散並列型動的解析システムは、現バージョンとなるまでに、開発期間として約4年を費やしている。その期間、システムが不安定になったり、並列処理による解析時間の低減率が悪かったりと、なかなか当初の目的をクリアすることができなかったが、長い時間と労力を投入して数値解析方法を改善し、またデータ転送量が最低となるようにシステムを改良し、現在に至っている。特に大きな改良点として、従前のシステムでは公開されていた通信用ライブラリーWMPI を用いて並列システムを構築していたが、Windows 用のソケット技術を用いて新たに通信用ライブラリーを開発し、これをシステムに組み込んだことが挙げられる。これにより通信速度も改善し、システムが不安定になる事はなくなった。

このマニュアルは、現時点 Ver.1.10 に関する解説書であるが、今後、改良・改善が加えられ、変更されることになる。システムを改良する上で、このマニュアルが役立つことを期待する。

ここでは、まず大規模数値シミュレーションを必要とする現状について考察しよう。構造設計における性能規定の導入によって、構造物の崩壊挙動や地震時の弾塑性応答に基づいた設計評価が、より重要となってくる。これに伴い、数値解析の需要は益々増加し、構造設計分野の数値解析でも、

さらに並列化効率を上げるためにスレーブの制御方法に変更を加えた。これを、Ver.1.11 とし、変更部分については、欄外および第4.4.7 節に示すことにする。

- 1) 構造物の大規模化に伴う自由度の増大
- 2) 弾塑性解析や座屈解析などを考慮し得る複雑なモデル化
- 3) 応力と歪の関係など構成則の高精度化
- 4) 動的応答解析での微小時間刻みによる繰返し回数の増加

等の理由で、大規模化していくものと考えられる。例えば、動的解析では数千自由度規模の解析モデルに対して、数万回の繰返し計算が必要となる。このような数値解析は、合理的な構造設計を実現するために必要不可欠であり、そのため、大規模数値解析を効率的に処理する方法を提案し、数値計算システムを開発することが今後の大きな課題となろう。

計算機の性能向上は著しいものがあり、現在の高性能パーソナルコンピュータは、一昔前の大型コンピュータに匹敵するほどの高速演算を行う。しかし、数値解析規模の拡大は現在もなお続いており、いつの時代においても解析システムの限界に迫る問題設定が成されるのが実情である。今後、構造解析に要求される条件を考えると、更なる大規模構造解析の実現が望まれており、現状の解析システムの能力では不十分となることは容易に予想される。

設計の合理化や省力化に伴い、実験的手法による評価はコスト的にも困難になっているのが現状であり、その代替手段として高性能コンピュータを用いた数値実験、数値シミュレーションが注目されている⁹⁾。大規模数値解析を行うコンピュータシステムとして、これまではベクトル型スーパーコンピュータを用いるのが一般的であり、これらを利用した建築構造解析は、和田ら¹⁰⁾が報告している。また、機械工学分野では、名取・野寺らが、研究報告¹¹⁾としてまとめている。これらは、いずれもスーパーコンピュータを必要とし、多くの人がその恩恵にあずかれるわけではない。一方、パーソナルコンピュータ（以下PCと略記）の性能向上とネットワークの普及により、スーパーコンピュータに替わる新しい高性能コンピュータシステムとして、分散並列処理システムが出現しようとしている。例えば、機械工学分野での矢川ら¹²⁾のように、その適用の可能性を模索して、様々な分野で研究・開発が試みられている。今後、PCによる分散並列処理を応用した数値シミュレーション技術の発展が大いに期待される。

数値計算で長時間かかる問題として、未知数が極端に多い場合（未知数10万元以上）と未知数が小規模でも（未知数数千元から一万元前後）繰返し計算が非常に多い場合とが存在する。無論、両者を含む問題はさらに時間がかかることになる。後者では、1増分処理の計算時間は短くても、この増分処理を繰返す数万ステップの計算では、膨大な計算

時間を必要とする。一般に、並列処理は前者の問題に適しており¹²⁾、現在も多くの研究が成されている。一方、動的問題に見られるような後者の問題は、並列処理には適していないとされており、ほとんど研究が進んでいない。その理由は、未知数が小規模で、サブストラクチャーに分割して計算する手法や他の並列技術を適用したとしても、多くの情報授受が各増分時間に発生するため、実行効率は上がり、むしろ単一計算より計算時間を長くすることさえあるためである。

そこで、SPACE 開発プロジェクトチームは、あえてこの問題に挑戦し、動的問題の並列処理化を試みることにした。現在、プロトタイプが開発され、処理効率やシステムの安定性に関する調査・研究を行っている。今後も、この分散並列型動的解析システムに関する研究・開発を継続して行っていく。

近い将来、実用的な分散並列型動的解析システムが SPACE に組み込まれ、利用可能となることを目指す。また、SPACE と同様に、このシステムもインターネット上で公開する予定である。

分散並列型動的解析の数値計算手法として、一般に並列処理に適していない陰解法を避け、反復解法を並列処理に適合するように改良して用いる。特に、ここでは、この反復解法が分散並列処理向きであることを示し、その特徴と処理能力の効率化について分析する。この解析手法は、汎用性があり、有限要素法によるソリッドモデルやシェル構造にも適用可能である。ただし、開発を目指している分散並列型構造解析システムは、線材による立体骨組構造を対象としている¹⁶⁾。

1.2 この本の目的と概要

このマニュアルは、分散並列型動的解析に関する仕様書であると同時に分散並列型の動的解析を学ぶ学生にとっての学習書でもある。このマニュアルは、大きく2つに分かれており、前半は分散並列型数値解析について、後半は独自に開発した並列型通信ライブラリーについて書かれている。以下では、各章において説明・解説する項目の概要を示す。

第2章では、動的解析における非線形方程式の数値解析手法を解説する。SPACE では、非線形振動方程式を効率良く数値解析するために、多くの工夫が成されているが、そのひとつとしてニューマーク 法を利用した反復解法を用いている。ここでは、その反復解法について説明し、どのように数値解析するかについて詳細に述べる。特に、この反復解法が分散並列処理に適していることを解説する。

後半部分では、分散並列型動的解析システムについて解説する。単一

CPU システムにおける動的解析フローを分析し、どの部分が分散並列処理に適しているかを検討する。さらに、これらの検討結果を元に構築した分散並列処理システムについて解説する。

分散並列処理システムは、今までに何度かの改良を加えている。特に、大きな改良点は、通信ライブラリーとして、WMPI から独自に開発した MPI_c に変更したことである。ここでは、それらの改良点について説明し、さらに並列処理の効果を評価し、解析時間の低減率を検討する。

第 3 章では、SPACE 分散並列型動的解析システムにおける PC クラスタ構築と動的解析ソルバーについて解説する。分散並列型の動的ソルバーは、基本的には単一 CPU のソルバーと同じであり、並列処理用に変更されている。また、同ソルバーは、マスター用とスレーブ用に分かれており、各々異なったプログラムとなっている。ここでは、2 つのソルバーについて、実際のコードを用いて解説する。

第 4 章では、分散並列型動的解析システムで使用しているマスターとスレーブ間のデータ転送システムについて解説する。マスターとスレーブ間のデータ転送には、多くの異なった種類があり、それぞれに対応してプログラムが開発されている。転送する種類としては、コントロールデータ、解析に必要な構造データなど、さらに、動的解析中で情報を共有するための加速度データ、応力データ、右辺項データなどである。これらについてもプログラムコードを用いて詳細に説明し、さらに転送するデータを具体的に表示することで、送信・受信の正確さを示す。最後に、並列用に拡張したサブルーチンについて解説する。

第 5 章では、独自に開発した並列用通信ライブラリーの仕様と使用法について説明する。このライブラリー MPI_c は、規格である MPI の一部の機能を実装したものであり、初期化、終了、送受信、状態取得などの機能を有する。このライブラリーの開発環境、あるいは、その仕様と使用法について説明する。第 6 章では、ライブラリー MPI_c に関する全プログラムを掲載し、その内容を説明する。

第 7 章では、mpi_c デーモンについて解説する。このデーモンは、分散並列解析を実行する際、既にスレーブ用の PC で起動されているものとする。デーモンは、マスター側からのアクセスによって、起動させるプログラム名と数項の情報を受け取り、その後、スレーブ用プログラムを立ち上げ、このスレーブに受け取った情報を渡す。ここでは、これらの仕事の内容と、デーモン用のプログラムの全てについて説明する。

第 8 章では、システム SPACE に対応した変更処理について解説する。分散並列型動的解析システムは、SPACE の動的解析用の仕様をそのまま

受け継いでおり、SPACE が変更されると、この分散並列型動的解析システムも再コンパイルすることで自動的に更新するように、開発環境が設計されている。ただし、新たに開発した並列用プログラムや、特に並列用書き直した SPACE 動的解析用プログラムは、自動的に変更されないで手動で更新する必要がある。ここでは、SPACE のバージョンアップに対応した分散並列型動的解析システムの変更方法を、具体的に説明する。

最後に、付録としてマスターとスレーブの主サブルーチンを掲載する。また、主文の中で説明できなかった他の並列用サブルーチンを全て記載する。

このマニュアルでは多くの技術用語が使用されている。その中でも特に、部材に関する用語は似ており、その意味を誤解し易い。そこで、その使用法について定義しておくことにする。

まず、動的解析対象となる構造物を解析モデルと呼ぶ。また、弾塑性解析を行うために各種の履歴モデルを有する部材が設計されており、これらを総称して部材モデルという。

リファレンスマニュアルに見られるように、入力時で要素と部材という用語を使い分けている。この使い分けは、入力時の手間と解析システム内の記憶領域を少しでも効率的にするためである。つまり、解析モデル内の部材全てが異なった断面を使用することはまれであり、そこで、同じ断面を有する部材を集めて、その部材全体に対して断面特性を定義するわけである。この集合の名称を要素として定義し、部材とは異なった使い方をする。例えば、要素数とは解析モデルの中で異なった断面を有する部材の数を表し、一方、部材数とは解析モデルにおける部材総数を表す。

SPACE ではこの要素に関連する情報を保存する構造体を Element と称し、また部材に関連する情報を保存する構造体を Member という。一般に Element は断面特性などのデータを保持し、解析中その内容は変化しない。一方、Member は、解析モデルの中の部材に特有の情報を保持し、また、計算に必要なワーク領域を提供する。そのため、構造体の成分では解析中その内容が変化するものが多い。

システム内で使用する部材モデルとして静的縮合モデルがある。ここでは、部材内部の節点によって部材が分割され、その分割された部材が異なった部材モデル、例えばファイバー断面や塑性論アナロジーモデル

1.3 SPACE で使用する用語について

などを持つことができる。これ以降、この静的縮合モデル内部の部材をエレメントと呼ぶ。また、上記した要素と部材を使い分けたように、このエレメントに対しても情報設定や記憶領域の省力化のために、この用語を用いて、要素内エレメントと部材内エレメントという用語を用い、異なった意味を持たせることにする。要素内エレメントとは、先に定義した要素内部に含まれる部材であり、部材内エレメントとは、部材内部に含まれる部材を指す。これらを区別する必要がない場合は、単にエレメントと呼ぶことにする。

このマニュアルでは、情報関連のカタカナ用語を少し用いている。意味があいまいにならないためにも、その言葉の使用法について説明する。まず、「コールする」あるいは「呼ぶ」は、サブルーチンや関数を呼び出し、制御をそちらに移すことを云う。次に、「インクルード」は、ヘッダーファイルなどをそのプログラム内に取り込み、そこで、そのファイルを展開することを云う。つまり、インクルードする側とされるファイルが同時にコンパイルされることになる。「ユーザー」とは、このSPACEを利用する人たちを指す。「システムハング」、「フリーズ」は、ほとんど同じ意味で使用する。前者は、アプリケーションプログラムの影響でOSそのものも止まってしまうことを指し、再起動しなければシステムが使用出来ない状態となる。後者は、SPACEプログラムが暴走または止まってしまう、応答を全くしない状態を指す。

プログラムの中で使用する「ゼロクリア」は、初期設定で使用する操作で、ある領域全体に0を挿入することであり、特に動的領域を確保しただけでは、その領域がどのような状態になっているか不明である。そこで、このようなゼロクリアを実行するわけである。次に、「コメント」は、プログラム内にコードや仕様などを書いておくコードであるが、コンパイル時には無視される。SPACEでは、このコメントを多く使用して、プログラムを読み易くしている。FORTRANとC++とでは、コメントコードの仕様が異なっているので注意されたい。

「メモリーリーク」とは、プログラム内で動的に領域確保が行われた後、解放を行わないままプログラムが終了すると、OSは、未だ該当するメモリーが使用されているとして、他で利用できる状態に復帰させることができないことを云う。そのため、OSを再起動しない限り、このメモリーは使用できなくなる。「マルチスレッド」とは、ひとつのプログラムの中で2つ以上の並行処理が行われることを云うが、ここでは、SPACEの中で数値計算と各種管理と図形処理を同時並行に処理することを云う。

本節では、SPACE Ver.3.00 に組み込まれている分散並列型動的解析システム（以後、並列システムと略記）の設定方法、及び実行方法について説明する。分散並列システムは動的解析のみ使用可能であり、親装置である Master とそれに制御される従属制御装置 Slave を用いて、大規模数値解析を効率的に処理するものである。さらに、Master、Slave とともに通常使用しているパーソナルコンピュータ（以下、PC と略記）で役割を果たすことができる。

最初に、SPACE Ver.3.00 を並列処理を行う全ての PC（Master、Slave）にインストールする。ただし、これらの PC は同じネットワーク環境に存在していなければならないことに注意されたい。

分散並列システムを構築するために、Master から次々と Slave となる動的ソルバーを立ち上げる必要がある。そこで、他の PC からの接続を待ち受ける状態にするため、mpi_c デーモンを全ての PC に立ち上げ、常駐させておかなければならない。そのため、mpi_c_deamon^{*1} をダブルクリックすることで、デーモンを手動で立ち上げて良いし、スタートアップに登録して、Windows 起動時に自動的に立ち上がるようにしておいても良い。このソフトは、非常に小さいプログラムなので、常駐していても他のシステムに影響を与えることはない。

デーモン起動（常駐）中は、画面下のタスクバー右のパレットにアイコンが表示される。このデーモンを終了させるためには、パレット内のアイコンを右クリックし、プルダウンメニューを出現させ、「ソフトの終了」を選択する。これで、デーモンを終了させることができる。



まず、Master となる PC に SPACE^{*2} とデーモン^{*1} を立ち上げ（インストール時にデスクトップにショートカットが作成される）SPACE よりコントロールファイルの新規作成を行う。

1.5 並列システムを用いた動的解析

1.5.1 分散並列システムの設定

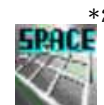
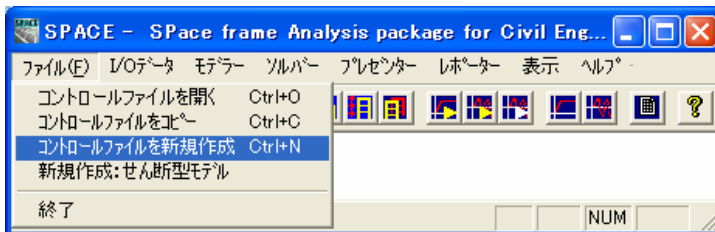


図 1-1 システムメニューによるコントロールファイルの新規作成

分散並列システムを用いた動的解析を行うための各種ファイル、及びデータの設定方法は、通常の設定方法と同様である。詳細はリファレンスマニュアルを参照されたい。なお、Slave となる PC はデーモンだけ立ち上げれば良い。

SPACE における分散並列システムを用いた動的解析では、3 種類の解析法が用意されている。以下にこれらを示す。

幾何学的非線形弾性解析
弾塑性解析
幾何学的非線形弾塑性解析

ただし、幾何学的非線形弾性解析、弾塑性解析、幾何学的非線形弾塑性解析を行うには、まず Master となる PC で固有値解析を行う(図 1-2 参照)必要がある。

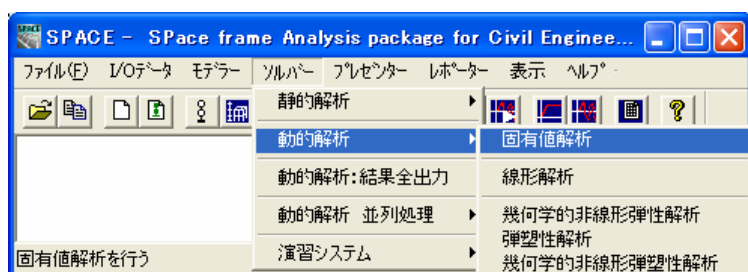


図 1-2 システムメニュー
による固有値解析実行

その理由は、レーリー減衰を利用するためには、固有振動数や振動モードが必要となるためである。固有値解析の詳細はリファレンスマニュアルを参照されたい。

固有値解析の正常終了を確認した後、次に分散並列処理による動的解析を実行する。まず、分散並列処理システムを起動するために、SPACE のメニューより、「ソルバー」「動的解析 並列処理」「幾何学的非線形線形解析 (任意の解析法)」を選択する。

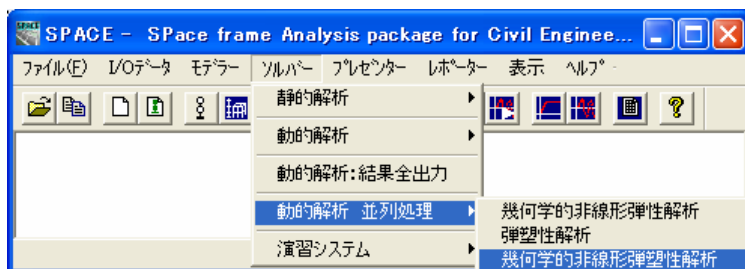


図 1-3 メニューにより分散並列処理を起動する

これで、分散並列動的解析システムが立ち上がることになる。後は、並列処理のための設定を行うことになる。

1.5.2 解析種別 及び固有値解析

1.6 解析実行 1.6.1 ソルバー を立ち上げる

並列分散処理システムが起動すると図1-3に示す動的解析用画面が表示される。ここでは、既に、メニューなどを用いて解析用モデルを表示している。次に、Master と Slave 用の PC を設定するために、メニューより「操作」「スレーブの選択」を選択して、設定用ダイアログを表示させる。

1.6.2 Master・Slave 用 PC の選択 及び設定方法

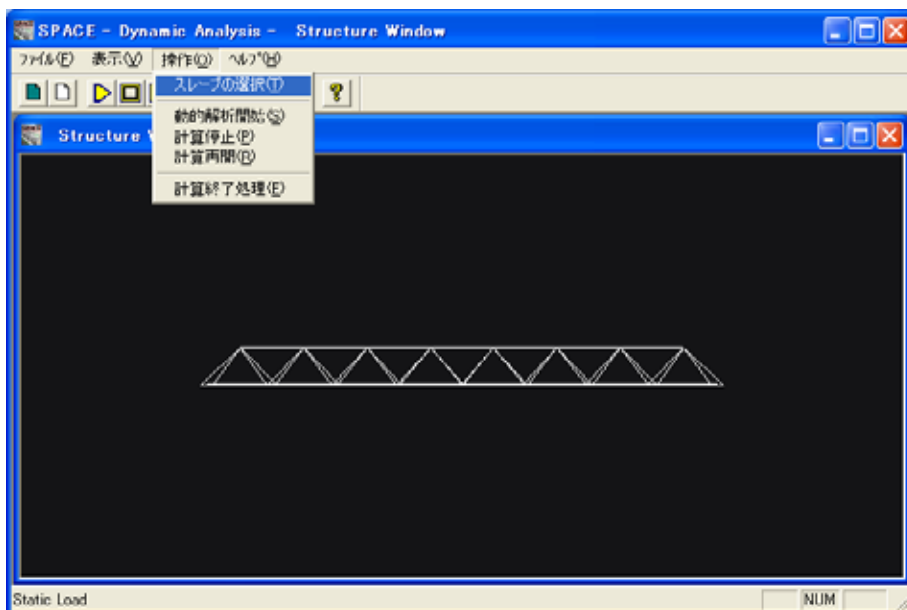


図 1-3 分散並列用解析画面と「スレーブの設定」メニューの選択

メニューによって表示されたダイアログ「Setup of PC Cluster」で Master と Slave の選択を行う。



図 1-4 「PC Cluster
の設定」ダイアログ

最初に、分散並列システムを行うためにクラスターを構築する。まず、ダイアログ下にあるプロセス設定項目において、「コンピュータ名 (IP アドレス)」に、Master または Slave となるコンピュータ名 (IP アドレス) を入力、または参照をクリックして選択する。次に、「プロセス ID」の「マスター」、または「スレーブ」を選択し、最後に担当部材番号を入力する。全てを設定した後、「追加」ボタンをクリックすると、プロセスリストに各設定が表記される。

続いて、同様の方法でクラスターを構成する全てのPCを設定し、リストに登録する。設定上の注意点は、リストの一番上をMasterとなるコンピュータとして (MasterとなるPCを最初に選択して設定を行う) Masterの担当部材番号を1~任意の数値に設定することである。また、担当部材番号の欠番、及び重複は解析エラーとなり、さらにはシステムダウンの原因となることもあることから注意が必要である。

担当部材の「分割設定」については、ユーザーが任意で設定を行っても良いが、全てのコンピュータを選択した後、ダイアログ内の「等分割」、または「クロック」ボタンをクリックして行うことも可能である。

「等分割」を選択すると、各コンピュータの担当部材数は均等に設定される。ただし Master となるコンピュータには、Slave の担当部材数に比べて 10% 少ない部材数が設定される。

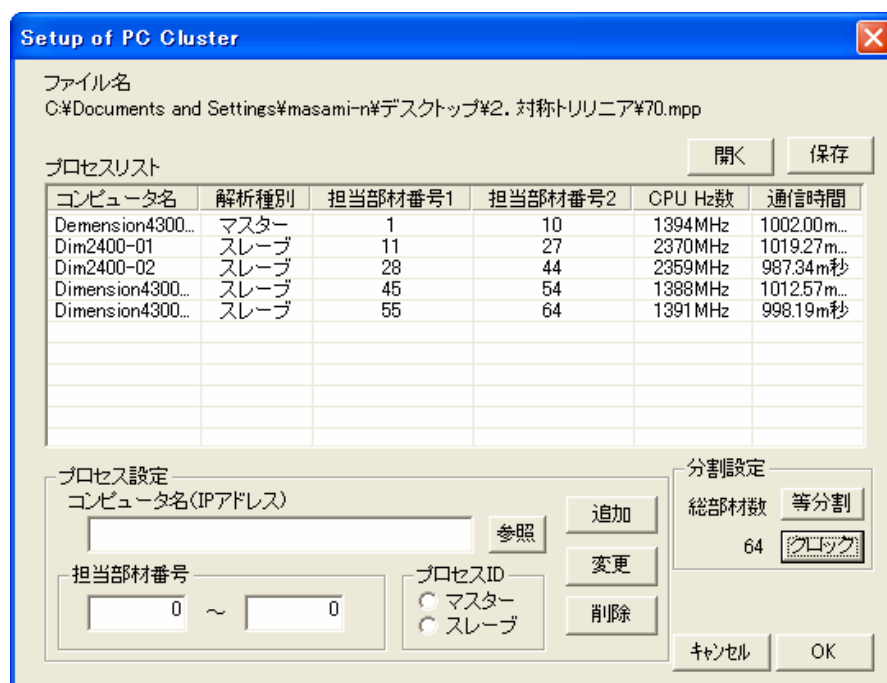


図 1-5 クロック
による担当部材の
設定

「クロック」を選択すると、各コンピュータの CPU の周波数と通信時間を計測して、その能力に応じて担当部材数の設定が行われる。その際、

クラスター内の全コンピュータのデーモンが立ち上がっていないと
ならない。

設定の変更を行いたい場合、まず、「プロセスリスト」内のコンピ
ュータをクリックして選択する。これで、「プロセス設定」内にその指定
したコンピュータのデータが表示され、その内容をここで書き換え「変
更」ボタンをクリックする。これで変更した内容がリストに再表示され
ることになる。

設定を削除したい場合、変更と同様に「プロセスリスト」内のコンピ
ュータを選択して、「削除」ボタンをクリックする。担当部材設定後に
変更、及び削除を行った場合、担当部材番号の欠番、及び重複に注意し
なければならない。またクロックによる分割設定は、訂正後にもう一度
行わなければならない。

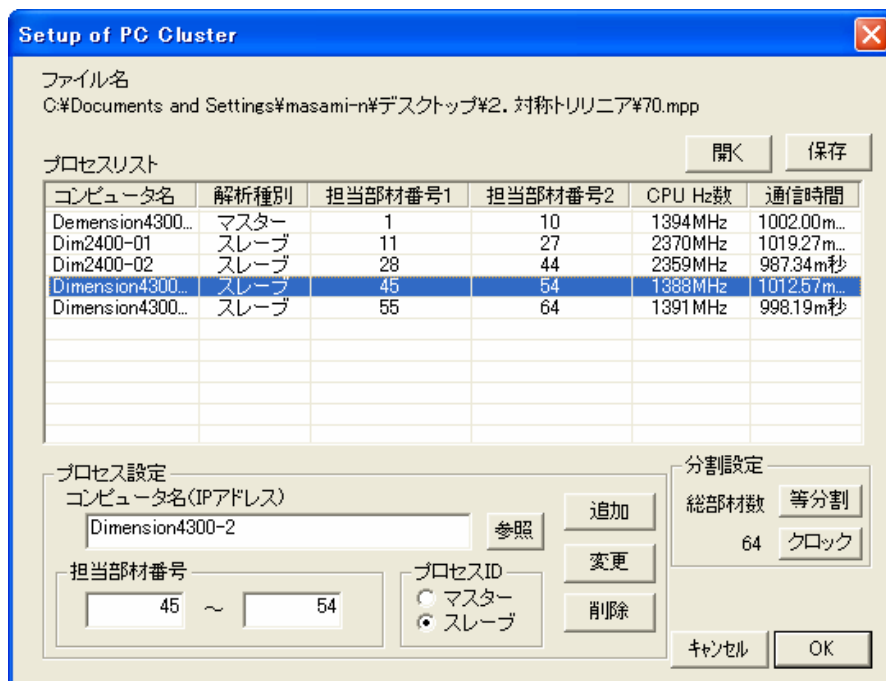


図 1-6 設定リス
トの変更

全ての設定が終了した後、ダイアログ内の「保存」をクリックし、図
1-7 の保存ダイアログを表示させ、ここでファイル名をつけて設定の保
存を行う。最後に、図 1-6 のダイア
ログで「OK」ボタンをクリックしてダイ
アログを閉じる。保存した設定は、次
に解析を行う際「開く」ボタンをクリ
ックして、このファイルを選択すると、
同様の設定を使用することができる。

図 1-7 クラスタ
ー情報の保存

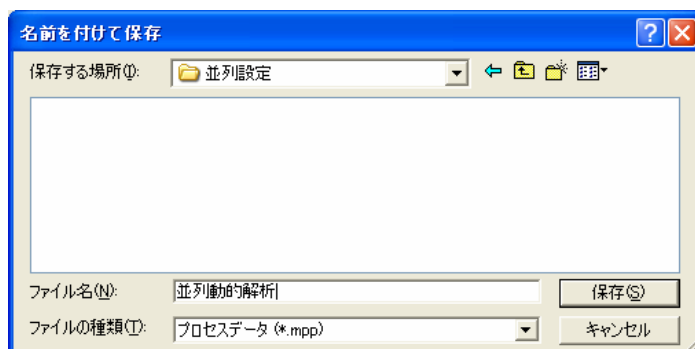


図 1-8 に示す動的ソルバー画面の実行ボタン*3をクリックすると、Slaveに次々とソルバーが立ち上げられ、解析が実行される。

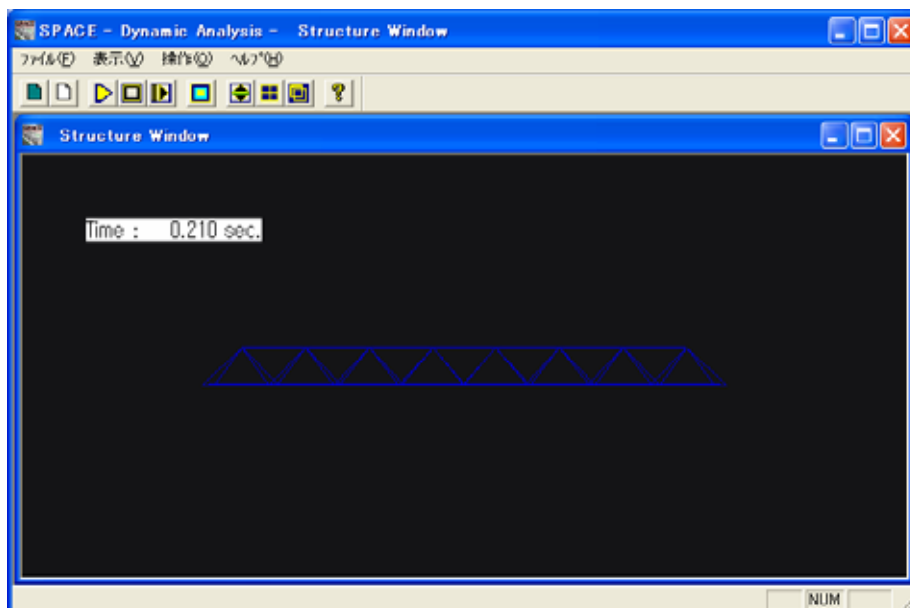


図 1-8 動的解析用分散並列処理が実行される

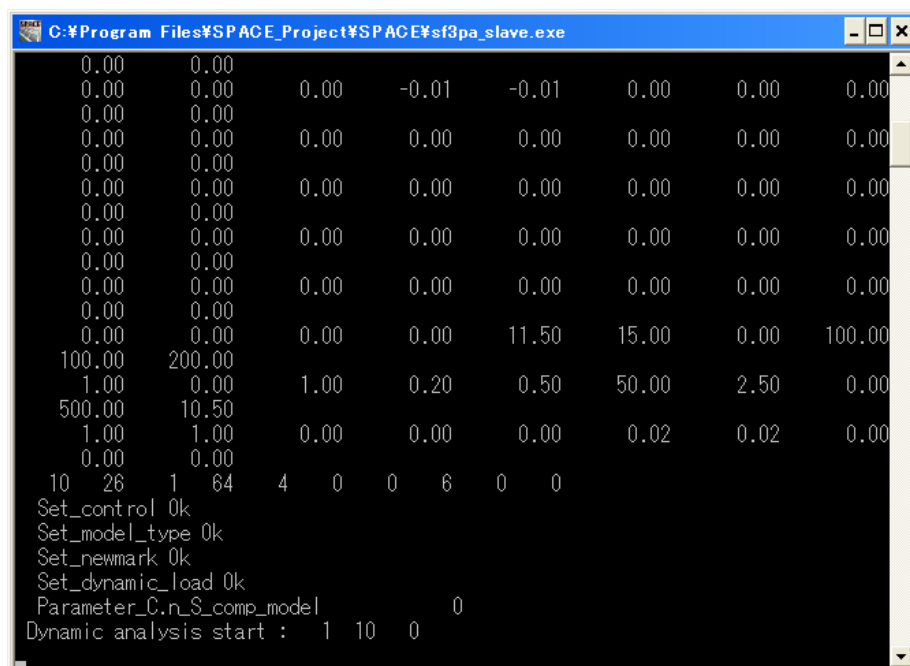


図 1-9 Slave の解析画面

解析が終了した後、Masterの動的ソルバー画面を閉じ、後は、SPACEのプレゼンター、及びレポーターを用いて解析結果の分析・表示、出力を行う。なお、Slaveのソルバーは解析終了後、自動的に閉じる。解析中にSlaveを閉じると、エラーが生じるためSlave処理を途中で絶対に中止しないよう注意されたい。